学考指要丛书
XUEKAO ZHIYAO CONGSHU

学士版

本章综述 释疑解难 典型例题 习题精解 综合练习

数字电子技术

学考指要

李云 编

西北工业大学出版社

数字电子技术学考指要

主编 李 云
编者 李 教 许 柯 孟 涛

西北工业大学出版社

【内容简介】 本书根据阎石主编的《数字电子技术基础》(第四版)内容编写而成。全书每章按照综述、释疑解难、典型例题、考研指点、综合练习5部分来编写。本书旨在帮助读者掌握课程重点内容,学会电路的分析设计方法,提高解题能力,为学习本课程及考研的读者提供帮助。

本书可供使用阎石主编的《数字电子技术基础》(第四版)和王毓银主编的《数字电路与逻辑设计》(第三版)教材的读者和教师参考,也可作为考研的读者和使用其他教材的读者的学习参考书。

图书在版编目(CIP)数据

数字电子技术学考指要/李云主编.—西安:西北工业大学出版社,2006.6
(学考指要丛书)
ISBN 7-5612-2074-X

Ⅰ.数… Ⅱ.李… Ⅲ.数字电路—电子技术—高等学校—教学参考资料 Ⅳ.TN79

中国版本图书馆CIP数据核字(2006)第040626号

出版发行:西北工业大学出版社
通信地址:西安市友谊西路127号 **邮编**:710072
电 话:(029)88493844 88491757
网 址:www.nwpup.com
印 刷 者:陕西向阳印务有限公司
开 本:787 mm×960 mm 1/16
印 张:22.875
字 数:617千字
版 次:2006年6月第1版 2006年6月第1次印刷
印 数:1~4 000册
定 价:30.00元

前　言

数字电子技术是通信、自动控制、计算机技术等专业的必修课程之一，又是报考上述专业硕士研究生的专业基础考试课程。鉴于当今电子技术课程中普遍存在的“听课容易、做作业难”的现象，为满足广大读者的需要，应西北工业大学出版社的约请，我们编写了本书，供学习数字电路课程的大学本科、专科、有志于考研的考生以及从事数字电路教学的教师参考。

本书以国内使用较为广泛的面向21世纪的两本教材——阎石主编的《数字电子技术基础》(第四版)、王毓银主编的《数字电路与逻辑设计》(第三版)中的内容为参考，总结作者多年的教学经验，通过多种题型向读者介绍解题方法和解题技巧，启迪读者的思维方式和解题能力，加深读者对数字电路课程内容的理解和应用。

全书共分8章，每章均包含以下内容：

(1)本章综述

对本章的内容进行概述、归纳、总结，帮助学生梳理教学内容中的基本概念，器件的逻辑功能，电路的分析和设计方法，指出重点、难点和考点。

(2)释疑解难

通过对在解题中一些容易犯的错误进行纠正，答疑解惑，进行点拨。

(3)典型例题

精选了具有代表性的典型例题进行分析，加深读者对基本概念、基本分析方法和设计方法的理解，归纳出解题的基本方法和技巧，做到举一反三，触类旁通。

(4)考研指点

通过对近年来部分高校的考研题目的分析，使读者掌握一些综合性的题目的分析及设计方法，巩固和加深对基本概念的理解及运用，提高解决问题的能力。

(5)综合练习

根据课程考试和考研要求，精选并编写了适量的自测题目，并附有答案。读者可通过这些测试题，检测自己对所学知识掌握的程度。

书后还给出了两套课程考试真题和两套模拟考题，并配有答案。

本书由李云主编，李教、许柯、孟涛参与了书中部分内容的编写。

由于编者水平所限，书中难免出现错误，恳请广大读者批评指正。

编　者

2006年2月

目 录

第 1 章
逻辑代数基础

1.1 本章综述

1.1.1 数字电路与模拟电路的区别

按照处理信号的不同，通常将电路分为两大类：模拟电路和数字电路。对应这两大类电路，在电子技术领域中就出现了数字电子技术和模拟电子技术两大分支。这两大分支的工程性、实践性都很强，并有着许多共同的特点和理论、实践基础，所以被统称为“电子技术基础”，但是它们之间又有许多明显的区别。所以在学习本课程时，既要注意它们的共性，更要重视它们的区别，抓住数字电路的特点。

1. 数字信号与模拟信号

在时间上和数值上都连续变化的物理量，如时间、温度、压力等称为模拟信号。

在时间上和数值上都离散的、不连续的物理量，比如开关的闭合与断开、脉冲的有与无等称为数字信号。数字信号反映在电路上只有“0”或者“1”两种状态，而无大小之分。

2. 数字电路的特点

(1) 输入、输出信号均为离散的、不连续的信号，其状态要么是“1”，要么是“0”。

(2) 半导体器件工作在饱和区或截止区，即工作在开关状态，因此，晶体三极管或场效应管均作为开关元件使用。

(3) 它所要研究的主要问题是输入和输出之间的逻辑关系，而不是大小和相位关系。

(4) 使用的主要工具有逻辑(布尔)代数、卡诺图、真值表和状态转换图等。

1.1.2 数制与码制

1. 数制

数制是表示一个数字量的多位数码中，每一位的构成方法以及从低位到高位的进位规则。数字电路中常用的数制有十进制、二进制、八进制和十六进制。其各种进制的特点如表 1.1 所示。

2. 不同进制数的相互转换

对于同一个数来说，既可以用十进制数形式表示，又可以用二进制数、八进制数或十六进制数形式表示，它们之间存在内在联系，可以相互转换。

(1) 任意进制数$(N)_R$转换成十进制数。其方法为：采用按权展开相加法。即将任意进制数$(N)_R$按位权展开，再相加便可求得相应的十进制数。

(2) 十进制数转换成任意进制数$(N)_R$。其方法为：将整数部分和小数部分分别进行转换，然后再将它们

合并起来。

整数部分：除 R 取余数，先低位后高位。即对于整数部分，将十进制整数反复除以 R，依次记录余数，便可得到 R 进制整数部分的各位数码，特别注意先得到的余数是 R 进制整数的最低位。

小数部分：乘 R 取整数，先高位后低位。即对于小数部分，将十进制小数反复乘以 R，依次记录整数，便可得到 R 进制小数部分的各位数码，注意先得到的整数是 R 进制小数的最高位。

(3) 二进制数、八进制数、十六进制数之间的转换。其方法为：以小数点为界，分别向左、向右按照 3 位二进制数对应一位八进制数，4 位二进制数对应一位十六进制数的规则，按位进行变换。

1) 将二进制数转换成八进制数或十六进制数。对于给定的二进制数，以小数点为界，分别向左、向右每 3 位合一位或每 4 位合一位，不够 3 位或 4 位的，添 0 补足，即可得相应的八进制数或十六进制数。

2) 将八进制数或十六进制数转换成二进制数。对于给定的八进制数或十六进制数，以小数点为界，分别向左、向右每一位用等值的三位二进制数或四位二进制数表示，即可得相应的二进制数。

表 1.1　各种进制的特点

	十进制数	二进制数	八进制数	十六进制数
数符	0,1,…,9	0,1	0,1,…,7	0,1,…,9,A,B,C,D,E,F 其中，A～F 依次表示十进制数 10～15
进/借位规律	逢十进一 借一当十	逢二进一 借一当二	逢八进一 借一当八	逢十六进一 借一当十六
基数	10	2	8	16
位权值	10^i	2^i	8^i	16^i
按位权展开	$(N)_{10}=\sum_{i=-m}^{n-1}a_i10^i$	$(N)_2=\sum_{i=-m}^{n-1}a_i2^i$	$(N)_8=\sum_{i=-m}^{n-1}a_i8^i$	$(N)_{16}=\sum_{i=-m}^{n-1}a_i16^i$

3. 二—十进制码(BCD 码)

BCD 码是用 4 位二进制码的 10 种组合来表示十进制数 0～9，所以 BCD 码是用二进制编码的十进制数，而不是二进制数。

由于编码的规则不一样，因此对应着不同的 BCD 码。

(1) 有权码：每位编码有固定的位权值。常用的有权码有 8421BCD 码，5421BCD 码等。

例如，8421BCD 码是采用 4 位二进制数的前 10 种组合，其每一位的位权从左到右依次为 8,4,2,1。即用 4 位二进制数的 0000～1001 分别表示十进制数的 0～9，而其余 6 种组合 1010～1111 是不允许出现的伪码。

(2) 无权码：每位编码没有固定的位权值。常用的无权码有余 3BCD 码，余 3 循环码等。

例如，余 3BCD 码是用 4 位二进制数的 0011～1100 分别表示十进制数的 0～9，可以看成由每位 8421BCD 码加 3(0011) 得来，所以它是一种无权码。

1.1.3　逻辑代数及逻辑函数的简化

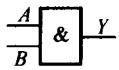
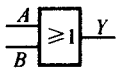
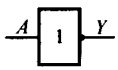
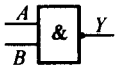
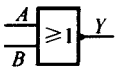
1. 逻辑代数与基本逻辑运算

逻辑代数是描述事物逻辑关系的数学方法，又叫布尔代数。其特点是：

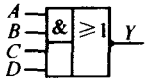
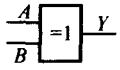
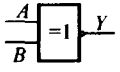
(1) 所有的变量和函数的取值范围只能是“0”和“1”两个值，因此这种变量与函数称为“逻辑变量”与“逻辑函数”。这里的逻辑值 0 和 1 表示互为对立的两种状态，如用 1 表示灯亮，则 0 表示灯灭；用 1 表示开关闭合，则 0 表示开关断开等。所以“0”和“1”本身没有大小之分。逻辑代数中的公式、规则、定理均要用二值逻辑的因果关系来理解。

(2) 基本的逻辑运算有与、或、非 3 种，对应的逻辑是与逻辑、或逻辑、非逻辑。利用这 3 种基本运算，可构成各种复合逻辑，如与非、或非、与或非、异或、同或等。实现这些逻辑运算的电路统称为门电路。为便于比较和应用，表 1.2 列出了描述 3 种基本逻辑和 5 种常用复合逻辑的逻辑符号、逻辑函数表达式、真值表和基本运算规则。

表 1.2　几种常用的逻辑

逻辑运算	逻辑符号	逻辑表达式	真值表	基本运算
与运算		$Y = AB$	A B | Y 0 0 | 0 0 1 | 0 1 0 | 0 1 1 | 1	$A \cdot 1 = A$ $A \cdot 0 = 0$ $A \cdot A = A$ $A \cdot \overline{A} = 0$
或运算		$Y = A + B$	A B | Y 0 0 | 0 0 1 | 1 1 0 | 1 1 1 | 1	$A + 1 = 1$ $A + 0 = A$ $A + A = A$ $A + \overline{A} = 1$
非运算		$Y = \overline{A}$	A | Y 0 | 1 1 | 0	$\overline{A} = A$
与非运算		$Y = \overline{AB}$	A B | Y 0 0 | 1 0 1 | 1 1 0 | 1 1 1 | 0	$\overline{A \cdot 1} = \overline{A}$ $\overline{A \cdot 0} = 1$ $\overline{A \cdot B} = \overline{A} + \overline{B}$
或非运算		$Y = \overline{A + B}$	A B | Y 0 0 | 1 0 1 | 0 1 0 | 0 1 1 | 0	$\overline{A + 0} = \overline{A}$ $\overline{A + 1} = 0$ $\overline{A + B} = \overline{A} \cdot \overline{B}$

续表

逻辑运算	逻辑符号	逻辑表达式	真值表	基本运算
与或非运算		$Y=\overline{AB+CD}$	A B C D Y　A B C D Y 0 0 0 0 1　1 0 0 0 1 0 0 0 1 1　1 0 0 1 1 0 0 1 0 1　1 0 1 0 1 0 0 1 1 0　1 0 1 1 0 0 1 0 0 1　1 1 0 0 0 0 1 0 0 1　1 1 0 0 0 0 1 0 1 1　1 1 0 1 0 0 1 1 1 0　1 1 1 1 0	$\overline{AB+CD}=\overline{AB}\cdot\overline{CD}$
异或运算		$Y=\overline{A}B+A\overline{B}$ $=A\oplus B$	A B Y 0 0 0 0 1 1 1 0 1 1 1 0	$A\oplus A=0$ $A\oplus\overline{A}=1$ $A\oplus 0=A$ $A\oplus 1=\overline{A}$
同或运算		$Y=AB+\overline{A}\,\overline{B}$ $=A\odot B$	A B Y 0 0 1 0 1 0 1 0 0 1 1 1	$A\odot B=\overline{A\oplus B}$ $A\oplus B=\overline{A\odot B}$ $A\oplus\overline{B}=\overline{A\oplus B}$ $A\oplus B\oplus C=$ $A\odot B\odot C$

2. 逻辑代数的基本定律、规则与定理

逻辑代数的基本定律、规则与定理是化简逻辑函数的基础，是逻辑运算的重要工具。

(1) 基本定律。根据与、或、非 3 种基本运算，可得到基本定律如表 1.3 所示。

表 1.3　逻辑代数的基本公式

名　称	表　达　式	说　明
0-1 律	$A+0=A$	变量与常量的关系
	$A\cdot 0=0$	
	$A+1=1$	
	$A\cdot 1=A$	
重叠律	$A+A=A$	逻辑代数特殊规律
	$A\cdot A=A$	
互补律	$A+\overline{A}=1$	
	$A\cdot\overline{A}=0$	
非非律	$\overline{\overline{A}}=A$	

续表

名　称	表　达　式	说　明
交换律	$A+B=B+A$	与普通代数规律相同
	$A \cdot B=B \cdot A$	
结合律	$(A+B)+C=A+(B+C)$	
	$(A \cdot B) \cdot C=A \cdot(B \cdot C)$	
分配律	$A \cdot(B+C)=A \cdot B+A \cdot C$	
	$A+BC=(A+B)(A+C)$	逻辑代数特殊规律
反演律(摩根定律)	$\overline{A+B}=\overline{A} \cdot \overline{B}$	
	$\overline{A \cdot B}=\overline{A}+\overline{B}$	

(2) 三个规律。逻辑代数的三个规律如表 1.4 所示。

表 1.4　逻辑代数的三个规则

规则名称	定　义	说　明
代入规则	任一个含有某变量的等式，如将所有出现该变量的地方都代之以一个函数，则等式仍成立	
对偶规则	对于任意一个函数表达式 F，如果将 F 中所有的“·”换成“+”，“+”换成“·”，“0”换成“1”，“1”换成“0”，但“原变量”和“反变量”都不变，并保持原来的逻辑优先顺序，则可得 F 函数的对偶函数 F'	① 为了保持原表达式的运算顺序，应适当地添加扩号 ② 在变换过程中应注意凡是在两个或两个以上变量上面的非号均应保持不变
反演规则	对于任意一个函数表达式 F，如果将 F 中所有的“·”换成“+”，“+”换成“·”，“0”换成“1”，“1”换成“0”，“原变量”换成“反变量”，“反变量”换成“原变量”，且保持原来的逻辑优先顺序，则可得 F 函数的反函数 $\overline{F}$	

(3) 基本定理。用表 1.3 中的基本定律，可推出表 1.5 所示的基本定理。

表 1.5　逻辑代数的基本定理

定理名称	表　达　式	含　　义
吸收定理	$A+AB=A$	在一个与或表达式中，若其中一项包含了另一项，则该项是多余的
	$A+\bar{A}B=A+B$	两个乘积项相或时，如果一项取反后是另一项的因子，则此因子是多余的
	$AB+A\bar{B}=A$	两个乘积项相或时，若两项中除去一个变量相反外，其余变量都相同，则可用相同的变量代替这两项
多余项定理	$AB+\bar{A}C+BC=AB+\bar{A}C$	若两个乘积项中分别包含了 A，$\bar{A}$ 两个因子，而这两项的其余因子组成第三项乘积项时，则第三个乘积项是多余的，可以消去

3. 逻辑函数及其简化

(1) 逻辑函数及其表示方法。逻辑函数反映的是实际逻辑问题中输入变量与输出变量之间的因果关系，可用逻辑函数表达式、真值表、卡诺图、逻辑图、波形图等方法来表示，如表 1.6 所示。

表 1.6　逻辑函数的常用表示方法

名　称	定　义	说　明
函数表达式	按照对应的逻辑关系，把输出变量表示为输入逻辑变量的与、或、非三种运算组合的表达式	同一个逻辑函数可以有不同形式的表达式，常见的有与或式、或与式、与非-与非式、或非-或非式和与或非式 5 种。各种形式表达式之间可以互相转换
真值表	将输入逻辑变量的各种可能取值和相应的函数值排列在一起而组成的表格	对于一个确定的逻辑函数，其真值表是唯一的。所以可利用真值表来检验两个看上去不一样的逻辑表达式是否相等
逻辑图	一种用逻辑符号构成的变量流程图，它可表示变量间的逻辑关系	由于逻辑函数的表达式形式不唯一，而不同形式的表达式可直接用相应的门电路实现，所以与其对应的就可能有多个逻辑图
卡诺图	是根据逻辑函数的真值表或最小项表达式按逻辑上的循环邻接特性画出来的一种方块图	由于真值表和最小项表达式均可以唯一地表示逻辑函数，所以卡诺图必然也可用来表示逻辑函数，并具有唯一性
波形图	指各个逻辑变量的逻辑值随时间变化的规律图，又叫时序图	

在各种表示方法中，对于同一逻辑函数，其真值表、卡诺图和波形图具有唯一性，而逻辑表达式和逻辑图

则具有多样性。逻辑函数表达式是逻辑函数最直接的表示方法，它可以通过建立真值表来获得。逻辑函数表达式有两种标准形式：一是最小项表达式，又叫标准与或式；二是最大项表达式，又叫标准或与式。这两种标准式对于给定的逻辑函数具有唯一性。

最小项与最大项：如表 1.7 所示为最小项与最大项的定义、表示方法、性质，以及最小项与最大项之间的关系。

表 1.7　最小项和最大项

	最小项(m_i)	最大项(M_i)	说　明
定　义	一个 n 变量的最小项，就是一个包含所有 n 变量的逻辑“与”项，其中每个变量都以原变量或反变量的形式出现一次，且仅出现一次	一个 n 变量的最大项，就是一个包含所有 n 变量的逻辑“或”项，其中每个变量都以原变量或反变量的形式出现一次，且仅出现一次	在提到最小项和最大项时，一定要说明变量的数目，否则最小项和最大项这一术语将失去意义。例如，乘积项 ABC 对三变量来说是最小项，而对四变量来说则是一般项；同样 $(A+B+C)$ 对三变量来说是最大项，而对四变量来说则是一般项
表示方法	通常用 m_i 来表示最小项，其中下标 i 叫序号。在变量按序排列后所组成的与项中，原变量用“1”表示，反变量用“0”表示，这 1 和 0 的顺序组成的二进制数对应的十进制即为 i 值。例如：3 变量最小项 $AB\overline{C} = m(110) = m_6$	通常用 M_i 来表示最大项，其中下标 i 叫序号。在变量按序排列后所组成的或项中，原变量用“0”表示，反变量用“1”表示，这 1 和 0 的顺序组成的二进制数对应的十进制数即为 i 值。例如：3 变量最大项 $(A+B+\overline{C}) = M(001) = M_1$	
性　质	① 对于变量的任一组取值，必有一个也只有一个最小项的值为 1； ② 对于变量的任一组取值，全体最小项之和恒为 1； ③ 对于变量的任一组取值，任意两个最小项的乘积恒为 0； ④ 两相邻最小项可以合并成一项，并可消去一个变量	① 对于变量的任一组取值，必有一个也只有一个最大项的值为 0； ② 对于变量的任一组取值，任意两个最大项之和恒为 1； ③ 对于变量的任一组取值，全体最大项之积恒为 0； ④ 只有一个变量不同的两个最大项的乘积等于各相同变量之和	
最小项和最大项的关系	① 对于 n 个逻辑变量的最小项和最大项，若其项号 i 值相同，则二者互补，可表示为 $m_i = \overline{M}_i$ 或 $m_i = \overline{M}_i$ ② 最小项与相应的最大项之和等于 1，即 $m_i + M_i = 1$		必须熟练掌握

最小项表达式：全部由最小项相或构成的与或式。

最大项表达式：全部由最大项相与构成的或与式。

任何一个逻辑函数，均可化成最小项表达式或者最大项表达式。

从最小项和最小项表达式的定义可以看出，实质上每个最小项都是对应于一种特定的变量组合，有多少

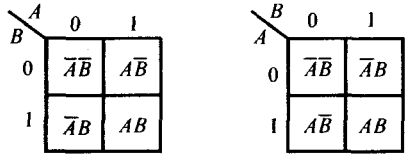
种不同的变量组合，就有多少个最小项。这与真值表是很类似的。真值表中对应于函数值为 1 的那些变量组合就相当于最小项表达式中存在的那些最小项；反之真值表中，对应于函数值为 0 的那些变量组合就相当于最大项表达式中存在的那些最大项。所以，真值表实质上就是一种最小项或最大项表，根据真值表中输出为 1 的项直接写出的逻辑表达式是最小项表达式；根据真值表中输出为 0 的项直接写出的逻辑表达式是最大项表达式。

(2) 逻辑函数的卡诺图化简。所谓卡诺图，就是变了形的真值表。

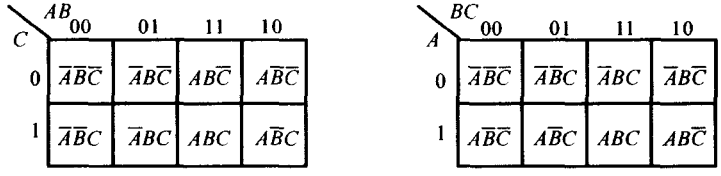
卡诺图的构成：对于 n 变量的逻辑函数，其卡诺图由 2^n 个小方格组成，每个小方格对应地表示一个最小项。

为了保证卡诺图中几何位置相邻的最小项在逻辑上也相邻，输入变量的取值组合必须排成循环码。所谓循环码是指相邻的两个码组之间只有一个位码元不同，如 $00 \to 01 \to 11 \to 10$，包括首尾也相邻。

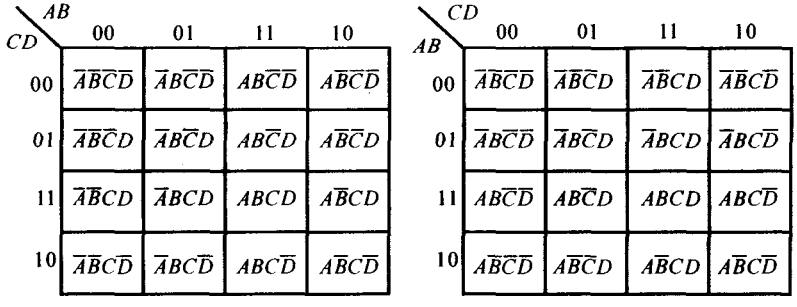
图 1.1(a)，(b)(c) 分别给出了 2 变量，3 变量，4 变量的卡诺图的常用形式。



(a)



(b)



(c)

图 1.1　2 变量，3 变量，4 变量卡诺图的常用形式

(3) 用卡诺图化简逻辑函数。在卡诺图中，逻辑相邻项合并规律是：只有 2^i 个相邻的小方格组成的方形才可圈圈合并为一项，合并的结果是消去不同(有 0，1 变化) 的变量，保留相同的变量。

1) 圈图原则：

① 被圈的小方格数必须是 2^i 个格。

② 每个小方格可以多次被圈，但每个圈内至少有一个小方格是初次被圈，否则会出现多余项。

③ 利用尽可能少的卡诺圈圈住逻辑函数的相关逻辑项，这样才能得出最简函数。

④ 在保证总圈数最少的前提下，圈要尽可能的大。

⑤ 求最简与或式：在卡诺图上通过圈 1 格合并化简。即在所圈的圈中，去掉不同的变量，保留相同的变量，且见 1 写原变量，见 0 写反变量，每个圈构成一个与项，将所有的与项相或，可得出原函数的最简与或式。

⑥ 求最简或与式：在卡诺图上通过圈 0 格合并化简。即在所圈的圈中，去掉不同的变量，保留相同的变量，且见 0 写原变量，见 1 写反变量，每个圈构成一个或项，将所有的或项相与，可得出原函数的最简或与式。

⑦ 当卡诺图中含有无关项时，为了获得最简结果，应充分利用无关项。根据化简需要，可把无关项任意看作“1”或“0”，即有利用函数的化简的无关项，化简时圈入卡诺圈，否则就不圈。

所谓无关项，它包括任意项和约束项。任意项是指在某些输入变量取值组合下，其函数值可以是 1 也可以是 0，并不影响电路的逻辑功能，这些变量取值所对应的最小项叫任意项。约束项是指在某些实际问题中，有些输入变量的取值受到约束不允许出现，那么它所对应的最小项叫约束项。

无关项在真值表或卡诺图中常用“d”或“×”表示。

通常把含有无关项的逻辑函数称为非完全描述的逻辑函数，而把不含无关项的逻辑函数称为完全描述的逻辑函数。

2）用卡诺图化简逻辑函数得到最简与或式的步骤：

① 将逻辑函数填入卡诺图中。

② 找出不能与其他“1”格合并的孤立“1”格，单独画圈。

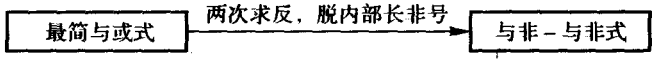
③ 找出只有一个合并方向的“1”格画圈，特别注意，圈应尽量的大。

④ 对有两种或两种以上圈法的“1”格画圈，特别注意，每个圈中至少要包含一个没有被其他圈圈过的“1”格；在总圈数最少的情况下，圈要尽量的大。

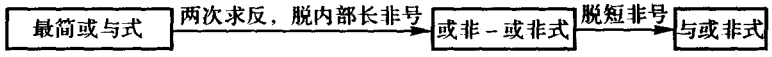
⑤ 由每个圈得到一个与项，与项中应保留相同（0，1 取值不变）的变量，消去不同（取值有 0，1 变化）的变量，再把所有与项相或即得到最简与或式。

4. 逻辑函数表达式形式的变换

由于通过逻辑函数的化简所得到的是最简“与或”表达式和最简“或与”表达式。在实际设计的数字系统中，为了减少所用器件的数目，提高带负载的能力，常用的是与非门、或非门、与或非门等，所以我们要将得到的最简“与或”式、“或与”式变换成与非-与非式、或非-或非式、与或非式等。其变换过程如图 1.2 所示。



(a)



(b)

图 1.2　逻辑表达式的变换

1.1.4 重点、难点及考点指南

1. 重点、难点

本章的重点是：

(1) 各种常用数制之间的相互转换。

(2) 基本逻辑运算及基本定律、规则与定理。

(3) 逻辑函数的 5 种(函数表达式、真值表、逻辑图、卡诺图、波形图) 表示方法及其之间的相互转换。

(4) 最小项和最大项的基本概念及两者之间的关系。

(5) 将任何一个逻辑函数表达式化成最小项表达式。

(6) 用卡诺图化简逻辑函数。

(7) 无关项的基本概念以及无关项在化简逻辑函数中的应用。

(8) 将最简与或式变换为与非-与非式；将最简或与式变换为或非-或非式。

本章的难点是卡诺图化简逻辑函数。

2. 本章的考点

(1) 常用的数制之间的相互转换。

(2) 常用码制之间，码制与数制之间的相互转换。

(3) 逻辑函数 5 种不同的表示方法。

(4) 基本逻辑运算及基本定律、规则与定理的应用。

(5) 逻辑函数的化简。

(6) 常见的题型：① 填空题；② 选择题；③ 判断或证明逻辑等式；④ 直接求逻辑函数的对偶式和反演式；⑤ 用卡诺图法化简逻辑函数(包含任意项)；⑥ 将一般逻辑函数式化成最小项表达式与最大项表达式；⑦ 求逻辑函数的与非-与非式和或非-或非式。

1.2 

问题 1.1 将十进制数 53.375 转换为等值的二进制数，其转换过程如下：

余数		整数	
2 | 53 …… 1		0.375	
2 | 26 …… 0		× 2	
2 | 13 …… 1		0.750	…… 0
2 | 6 …… 0		0.75	
2 | 3 …… 1		× 2	
2 | 1 …… 1		1.50	…… 1
0		0.5	
		× 2	
		1.0	…… 1

所以 $(53.375)_{10} = (101011.011)_2$，该解法对吗？

答：该题解法正确，但答案不对。

错误发生在整数 53 不断地除 2，先得到的余数为低位，后得到的余数为高位，所以十进制数 53 转换成二进制数时，对所求的余数应从下向上依次顺序写成 110101。正确的答案为 $(53.375)_{10} = (110101.011)_2$。

将十进制数转换为二进制数时，要将整数部分和小数部分分别进行转换，特别注意书写顺序由高位到低位，整数部分除 2 取余数，且先低位后高位；小数部分乘 2 取整数，且先高位后低位。

问题 1.2　将十进制数 37.625 转换为等值的十六进制数，其转换过程如下：

		余数		整数
16⌊37	……	5(低位)	0.625	
16⌊2	……	2(高位)	× 16	
0			3750	
			625	
			10.000	…… 10

所以(37.625) = $(25.10)_{16}$，问该解法对吗？

答：不对。

错误发生在小数 0.625 乘以 16 处，因为十六进制数的 16 个数符为 0 ～ 9，A ～ F，其中 A ～ F 分别表示十进制数的 10 ～ 15，所以十六进制的十要用 A 表示，而十六进制的 10 表示十六进制数的 16。所以正确的解法如下：

		余数		整数
16⌊37	……	5(低位)	0.625	
16⌊2	……	2(高位)	× 16	
0			3750	
			625	
			A.000	…… A

所以　　$(37.625)_{10} = (25.A)_{16}$

十六进制的 10 ～ 15 应分别用 A ～ F 表示，十六进制的 10 表示 16。

问题 1.3　将二进制数$(101101)_2$转换成相应的十进制数，其转换过程如下：

$$(101101)_2 = 2^6 + 2^4 + 2^3 + 2^1 = (90)_{10}$$

试问该解法对吗？

答：不对。

错误发生在把二进制数的每位的位权搞错。二进制数最低位的位权应为 2^0，所以二进制数每位的位权从低到高依次为 $2^0, 2^1, 2^2, 2^3, 2^4, 2^5, \cdots$。所以正确的解法是

$$(101101)_2 = 2^5 + 2^3 + 2^2 + 2^0 = (45)_{10}$$

将二进制数转换为十进制数时，其二进制数整数部分的位权从低位到高位依次为 $2^0, 2^1, 2^2, 2^3, \cdots$。特别要注意整数部分的最低位的位权为 2^0。

问题 1.4　“BCD 码是二进制数”对吗？

答：不对。BCD 码不是二进制数，而是用二进制编码表示的十进制数。

问题 1.5　若逻辑等式 $AC = AD$ 成立，则 $C = D$ 对吗？

答：不对。因为当 $A = 1$ 时，有 $C = D$；而当 $A = 0$ 时，则 C 和 D 不一定相等。